|  |
| --- |
| Изображение выглядит как зарисовка, рисунок, корона, символ  Автоматически созданное описание  МИНОБРНАУКИ РОССИИ |
| Федеральное государственное бюджетное образовательное учреждение высшего образования **«МИРЭА − Российский технологический университет»**  **РТУ МИРЭА** |

**Институт информационных технологий (ИИТ)**

**Кафедра инструментального и прикладного программного обеспечения (ИиППО)**

**ОТЧЁТ ПО ПРАКТИЧЕСКИМ РАБОТАМ**

**по дисциплине «Информационный менеджмент программных продуктов и систем»**

2024/25 уч.г.

**Наименование проекта-прототипа для проведения анализа и реинжиниринга информационного менеджмента**: «сервис планирования подарков “Wishlist”»

|  |  |  |  |
| --- | --- | --- | --- |
| Отчет представлен к  рассмотрению:  Студент гр. ИКБО-20-21 | «\_\_» \_\_\_\_\_\_\_ 2024 | (подпись) | Сидоров С.Д. |
| Отчёт принят:  Ассистент каф. ИиППО: | «\_\_» \_\_\_\_\_\_\_ 2024 | (подпись) | Братусь Н.В. |

Москва 2024

УДК 004.92

Сидоров С.Д., Отчет по практическим работам по дисциплине «Информационный менеджмент программных продуктов и систем» по образовательной программе «Разработка программных продуктов и проектирование информационных систем» направления подготовки 09.03.04 «Программная инженерия» М. 2024 г., МИРЭА – Российский технологический университет (РТУ МИРЭА), Институт Информационных Технологий (ИИТ), кафедра инструментального и прикладного программного обеспечения (ИиППО) - 18 стр., 2 рис., 1 табл., , 3 источн. , 1 прил.

Ключевые слова: информационный менеджмент, реинжиниринг, экспертная оценка, анализ, Wishlist.

Целью работы является анализ и реинжиниринг информационного менеджмента проекта-прототипа «сервис планирования подарков “Wishlist”»».

Sidorov S.D., Report on practical work in the discipline " Information management of software products and systems" under the educational program "Software product development and design of information systems" training directions 09.03.04 "Software Engineering" M. 2024, MIREA – Russian Technological University (RTU MIREA), Institute of Information Technology (IIT), Department of Tool and Applied Software (Department of TAS) - 18 pages, 2 figures, 1 table, 3 sources, 1 appendix.

Keywords: information management, reengineering, expert assessment, analysis, Wishlist.

The purpose of the work is the analysis and reengineering of information management of the prototype project "presents planning service “Wishlist”".

**СОДЕРЖАНИЕ**

[ПЕРЕЧЕНЬ СОКРАЩЕНИЙ 4](#_Toc177919853)

[ПЕРЕЧЕНЬ ТЕРМИНОВ И ОПРЕДЕЛЕНИЙ 5](#_Toc177919854)

[1 Выбор проекта-прототипа, составление и согласование перечня показателей анализа прототипа 6](#_Toc177919855)

[1.1 Описание проекта 6](#_Toc177919856)

[1.2 Критерии анализа 7](#_Toc177919857)

[2 Экспертный анализ проекта-прототипа 9](#_Toc177919858)

[3 Реинжиниринг информационного менеджмента проекта-прототипа и экспертная оценка модифицированной версии 11](#_Toc177919859)

[КАТАЛОЖНОЕ ОПИСАНИЕ 14](#_Toc177919860)

[СПРАВКА 16](#_Toc177919861)

[СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 17](#_Toc177919862)

[ПРИЛОЖЕНИЕ А 18](#_Toc177919863)

ПЕРЕЧЕНЬ СОКРАЩЕНИЙ

В настоящем отчете применяют следующие сокращения и обозначения.

|  |  |  |
| --- | --- | --- |
| БД | – | база данных |
| ИС | – | информационная система |
| ПО | – | программное обеспечение |
| СУБД | – | система управления базами данных |
| API | – | Application Programming Interface (прикладной программный интерфейс) |

ПЕРЕЧЕНЬ ТЕРМИНОВ И ОПРЕДЕЛЕНИЙ

В настоящем отчете применяют следующие термины с соответствующими определениями.

|  |  |  |
| --- | --- | --- |
| База данных | – | это структурированное хранилище данных, которое позволяет удобно хранить, искать и обновлять информацию. |
| Архитектура приложения | – | это набор принципов и правил для разработки, которые упрощают создание, поддержку и расширение приложения. |

1. Выбор проекта-прототипа, составление и согласование перечня показателей анализа прототипа
   1. Описание проекта

В рамках данного проекта необходимо разработать сервис планирования подарков «Wishlist», который представляет собой веб-приложение, которое позволяет пользователям создавать и управлять списками подарков, делиться ими с друзьями и резервировать подарки из вишлистов других пользователей. Приложение включает систему учетных записей, систему вишлистов и подарков, а также модуль резервирования подарков. Пользователи могут добавлять друзей, отправлять запросы на дружбу и назначать соавторов для совместного редактирования вишлистов. Присутствует поддержка уведомлений о днях рождения и других значимых событиях, а также автоматизация создания карточек подарков на основе введенных данных.

Сервис поддерживает бизнес-аккаунты, позволяя компаниям публиковать подборки товаров, продвигать свои продукты и размещать рекламные баннеры на платформе. Модуль идей для подарков предлагает пользователям просмотр публичных вишлистов и добавление понравившихся подарков в личные списки. Пользователи могут резервировать подарки, чтобы избежать дублирования, и получать уведомления о действиях друзей и соавторов.

Цели проекта включают упрощение процесса выбора подарков, снижение вероятности дублирования, поддержку бизнес-аккаунтов для продвижения товаров и увеличение вовлеченности пользователей. Основные задачи: предоставление удобного интерфейса для управления вишлистами и резервирования подарков, интеграция с социальными сетями, повышение безопасности данных, автоматизация тестирования и мониторинга. Проект имеет потенциал для дальнейшего развития за счет внедрения новых функций и интеграции с внешними системами.

Проект разрабатывается с использованием языков программирования TypeScript и JavaScript. Серверная часть реализуется на платформе Node.js с использованием фреймворка Nest.js, а клиентская часть — на библиотеке React. В качестве СУБД используется PostgreSQL. Архитектура системы строится по принципу клиент-серверного взаимодействия с применением REST API для обмена данными между клиентом и сервером. Клиентское приложение разрабатывается в виде SPA (Single Page Application) с использованием Redux для управления состоянием приложения. Реализуются современные методы безопасности, включая шифрование данных и JWT (Json Web Token) для аутентификации и авторизации пользователей.

Критерии анализа

Был выбран следующий перечень позиций экспертизы проекта:

1. **Устойчивость проекта:** оценка способности проекта адаптироваться к изменениям и продолжать функционировать в условиях непредвиденных ситуаций.
2. **Коэффициент запаса:** измерение степени безопасности и резерва ресурсов, заложенных в проекте для обеспечения его стабильного функционирования.
3. **Обеспечение жизненного цикла (ЖЦ):** наличие и качество механизмов для поддержки всех этапов ЖЦ проекта, включая разработку, развертывание, эксплуатацию и вывод из эксплуатации.
4. **Ресурсоёмкость:** оценка количества ресурсов (время, вычислительная мощность, финансовые затраты), необходимых для реализации и поддержки проекта.
5. **Информационная безопасность:** оценка мер и механизмов защиты данных пользователей и предотвращения несанкционированного доступа к системе.
6. **Коэффициент готовности:** степень завершённости проекта и готовность к развертыванию или эксплуатации.
7. **Информационное обеспечение мер по диагностике:** наличие и качество инструментов и механизмов для диагностики и мониторинга состояния проекта.
8. **Профилактические меры:** наличие и реализация мер для предотвращения потенциальных проблем и сбоев в проекте.
9. **Аварийно-восстановительные меры на всём ЖЦ:** эффективность мер по быстрому восстановлению проекта в случае возникновения сбоев или аварийных ситуаций.
10. **Оценка проектных рисков:** идентификация, анализ и управление рисками, связанными с реализацией и эксплуатацией проекта.
11. **Надёжность и нагруженность изделия:** оценка способности системы выдерживать нагрузку и обеспечивать стабильную работу без сбоев.
12. **Наличие и качество каталогизации изделия проекта:** наличие и организация структурированной информации о проекте и его компонентах для удобства поиска и поддержки.
13. **Эргономические составляющие:** оценка удобства использования интерфейсов и взаимодействия пользователей с системой.
14. **Технико-эстетические составляющие:** качество и привлекательность визуального и функционального дизайна системы.
15. **Юзабилити и пользовательский опыт:** уровень удобства использования системы конечными пользователями, включая оценку интуитивности интерфейсов и удовлетворенности пользователей.
16. Экспертный анализ проекта-прототипа
17. Устойчивость проекта – 4.

Проект имеет базовую устойчивость, позволяющую функционировать при средних нагрузках, но возможны проблемы при масштабировании.

1. **Коэффициент запаса – 4.**

Проект имеет базовую систему резервирования и распределения задач, но не предоставляет возможности расширения функционала без полной переделки архитектуры.

1. **Обеспечение жизненного цикла - 3**

Имеется поддержка ЖЦ в виде настроек учетных записей и возможностей изменения структуры данных. Отсутствуют описанные механизмы миграций и контроля версий на уровне базы данных.

1. **Ресурсоёмкость – 3.**

Приложение использует эффективные средства на клиенте и сервере, однако выбор инструментов (React, NestJS) предполагает значительную нагрузку на систему при большом количестве пользователей.

1. **Информационная безопасность – 4.**

Использование JWT и httpOnly кук для аутентификации соответствует современным стандартам безопасности. Возможны улучшения путем реализации двухфакторной аутентификации.

1. **Коэффициент готовности – 4.**

Проект реализован на высоком уровне, имеет основные функциональные возможности и архитектуру. Однако отсутствуют детали по тестированию и документации API.

1. **Информационное обеспечение мер по диагностике – 2.**

Отсутствует система логирования и мониторинга состояния системы.

1. **Профилактические меры – 2.**

Не описаны механизмы автоматической проверки состояния системы и предотвращения сбоев. Нет мониторинга на уровне инфраструктуры.

1. **Аварийно-восстановительные меры на всём ЖЦ – 2.**

Не описаны механизмы автоматической проверки состояния системы и предотвращения сбоев. Нет мониторинга на уровне инфраструктуры.

1. **Оценка проектных рисков – 3.**

Угрозы проекту включают риски сбоя связи и масштабируемости. Отсутствует стратегия управления рисками.

1. **Надёжность и нагруженность изделия – 3.**

Способность проекта выдерживать значительные нагрузки ограничена из-за отсутствия механизма балансировки и динамического масштабирования.

1. **Наличие и качество каталогизации изделия проекта – 3.**

Каталогизация существует в виде документации и комментариев в коде, однако отсутствуют инструкции по миграциям и развёртыванию.

1. **Эргономические составляющие – 4.**

Интерфейс проекта удобен, но требует адаптивного дизайна для различных типов устройств. Обновление интерфейса и улучшение адаптивности дизайна повысило удобство работы на всех типах устройств, включая мобильные платформы.

1. **Технико-эстетические составляющие – 3.**

Проект обладает устаревшим дизайном с не читабельными сочетаниями цветов, а также не корректными решениями в плане расположения элементов на экране.

1. **Юзабилити и пользовательский опыт – 5.**

Пользовательский опыт продуман, интерфейс интуитивно понятен, имеется возможность гибкой настройки и взаимодействия с сервисом.

Проект демонстрирует высокий уровень качества позиций, связанных с функционированием приложения с точки зрения пользователя, однако демонстрирует плохой уровень готовности к эксплуатации в продуктовой среде.

1. Реинжиниринг информационного менеджмента проекта-прототипа и экспертная оценка модифицированной версии

Для улучшения проекта-прототипа были предложены следующие решения:

1. Добавить Prometheus и Grafana для сбора метрик, ELK-стек для логирования и автоматические оповещения о сбоях.
2. Реализовать миграции с использованием Flyway, улучшить индексацию и рассмотреть репликацию для повышения отказоустойчивости.
3. Настроить GitLab CI или Jenkins для автоматического тестирования, сборки и развертывания.
4. Ввести двухфакторную аутентификацию (2FA) и регулярное тестирование безопасности.
5. Реализовать автоматическое резервное копирование базы данных и создать документированный процесс восстановления.
6. Внедрить Redis или Memcached для снижения нагрузки на базу данных и ускорения ответа на повторные запросы.
7. Использовать Nginx или HAProxy для распределения запросов и повышения стабильности системы.
8. Адаптировать UI под мобильные устройства и улучшить юзабилити на основе A/B тестирования.
9. Создать API-документацию на основе Swagger и добавить руководство по развертыванию и конфигурации.
10. Идентифицировать возможные риски, разработать меры смягчения и ввести процесс регулярного мониторинга.

В проекте присутствуют риски связанные с масштабируемостью системы, которая может быть не готова к резкому увеличению нагрузок и количества пользователей. Для смягчения предлагается использовать контейнеры для возможности в кратчайшие сроки развернуть новый экземпляр приложения и перенаправить пользователей.

Также необходимо учитывать возможность утери данных с сервера базы данных или его неработоспособность. Для минимизации вреда предлагается использовать регулярное бекапирование, а также поддерживать в рабочем состоянии реплику текущей базы данных, для возможности быстрого перехода на её использование в случае недоступности основного сервера.

Также следует учесть риски связанные с обновлением системы, при которых могут происходить сбои в работе и приложение, может быть, не доступно, что приведет к потере пользователей. В качестве решения предлагается использовать различные оркестраторы контейнеров, например Docker Swarm, что позволит в случае сбоя откатится на предыдущую версию приложения и сделать это незаметно для пользователя.

В таблице 1 представлены потенциальные результаты реинжиниринга прототипа.

Таблица 1 – Сравнение результатов реинжиниринга

|  |  |  |
| --- | --- | --- |
| **Критерий** | **Оценка до улучшений** | **Оценка после улучшений** |
| Устойчивость проекта | 4 | 5 |
| Коэффициент запаса | 3 | 4 |
| Обеспечение жизненного цикла | 3 | 5 |
| Информационная безопасность | 4 | 5 |
| Коэффициент готовности | 4 | 5 |
| Информационное обеспечение мер по диагностике | 2 | 5 |
| Профилактические меры | 2 | 4 |
| |  | | --- | |  |  |  | | --- | | Аварийно-восстановительные меры на всём ЖЦ | | 2 | 4 |
| |  | | --- | |  |  |  | | --- | | Оценка проектных рисков | | 3 | 5 |
| Надёжность и нагруженность изделия | 3 | 4 |
| |  | | --- | | Наличие и качество каталогизации изделия проекта | | 3 | 5 |
| Эргономические составляющие | 4 | 4 |
| Технико-эстетические составляющие | 3 | 4 |
| Юзабилити и пользовательский опыт | 5 | 5 |

КАТАЛОЖНОЕ ОПИСАНИЕ

По результату работы было составлено трёхзвенное проектное соглашение, а именно:

* языковое / платформенное (кроссплатформенное) соглашение проекта: в качестве языков программирования выбраны TypeScript для клиентской части (React) и JavaScript для серверной части (NestJS). Используемая платформа — Node.js. Проект кроссплатформенный, функционирует на любых ОС (Windows, Linux) в рамках контейнеров Docker. Это позволяет легко развёртывать и масштабировать проект в любых средах, включая облачные решения.
* онтологическое соглашение проекта: логическая структура системы построена на основе MVC (Model-View-Controller) и Flux-архитектуры для фронтенда. Основные компоненты включают: систему вишлистов и управления подарками, систему друзей и управления пользователями, систему уведомлений и резервирования. Для хранения данных используется PostgreSQL с использованием ORM Sequelize. Для обеспечения безопасности применяются JWT токены и httpOnly куки. Для хранения данных будет спроектирована БД по данной ER-модели (Рисунок 1).

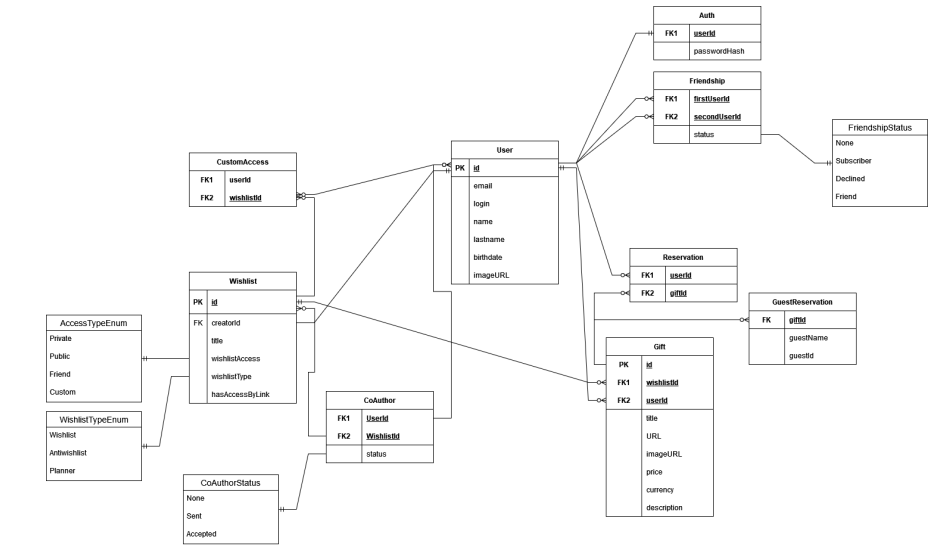


Рисунок 1 – ER-диаграмма БД

* управленческое соглашение проекта: методология разработки — Agile. Выбрана для обеспечения гибкости и быстрого реагирования на изменения в ходе разработки и тестирования функционала. Это необходимо для реализации итеративных изменений и добавления нового функционала в процессе реинжиниринга. Жизненный цикл проекта включает этапы проектирования, разработки, тестирования, развёртывания и поддержки, с выделением спринтов продолжительностью в 2 недели.

СПРАВКА

ПК-2.4 - Выполняет обработку информации с целью подготовки решений по концептуальному, функциональному и логическому проектированию клиент-серверных информационных систем

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Методические указания к выполнению практический работ [Электронный ресурс], метод. Указания / Р. Г. Болбаков, М. Ю. Волков, В. Т. Матчин, В. А. Мордвинов. — М.: РТУ МИРЭА, 2019. — Электрон. опт. Диск (ISO)
2. Аникеев Е.В. iOS-приложение для строительных компаний с использованием RTMP Образования // Выпускная квалификационная работа бакалавра / Федеральное государственное автономное образовательное учреждение высшего образования «Национальный исследовательский университет «Высшая школа экономики». - Москва, 2024. - 46 с. [сайт] — URL: https://www.hse.ru/ba/ami/students/diplomas/925074581 (дата обращения: 29.09.2024)
3. ГОСТ 7.32-2017. Отчет о научно-исследовательской работе. Структура и правила оформления. [сайт] — URL: http://www.lib.surgu.ru/media/files/gost\_7.32-2017.pdf (дата обращения 29.09.2024)

ПРИЛОЖЕНИЕ А

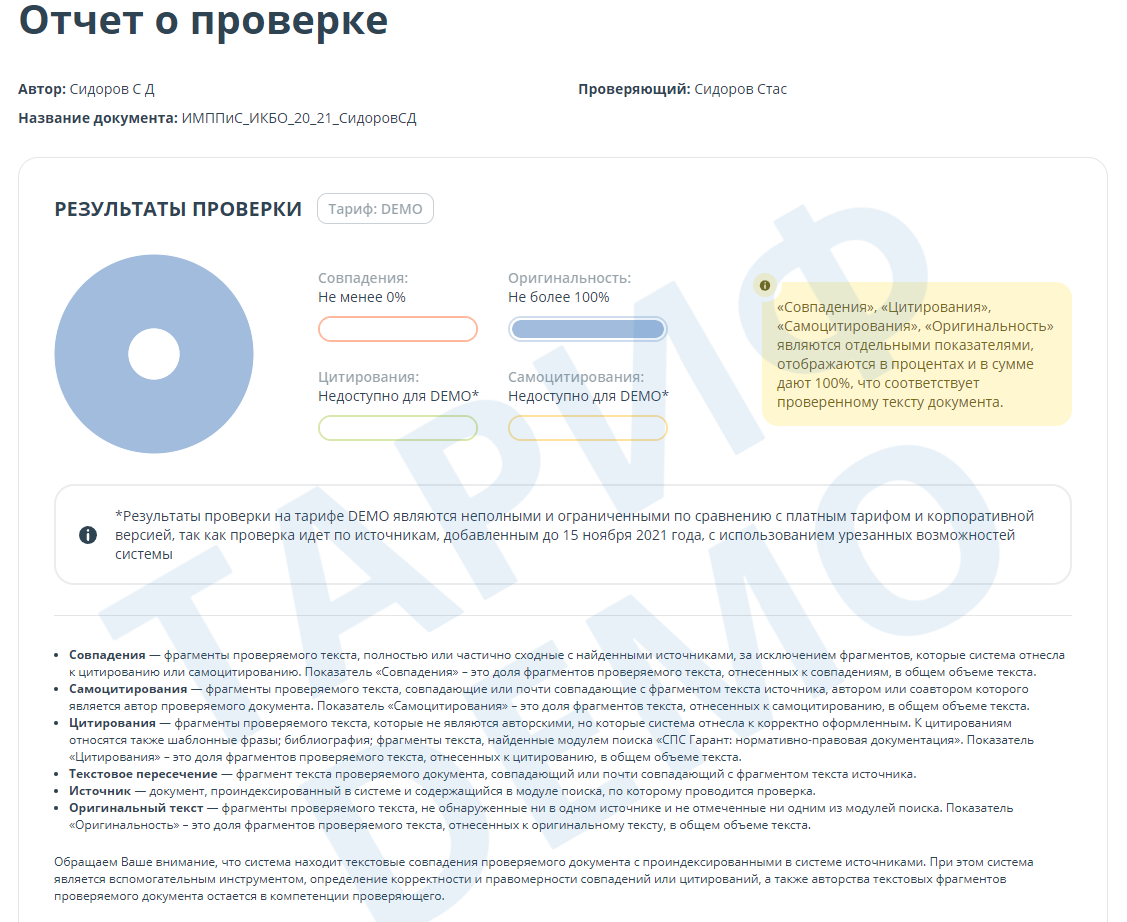


Рисунок А.1 – Отчет об антиплагиате